LAST NAME: ______________________

FIRST NAME: ______________________

Problem 1 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$b^i c^j d^k c^\ell b^m a^n d^p a^q$$

$b^m d^p b^m d^p$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = m,\ k = p,\ j = 0,\ q = 0,\ \ell = 0,\ n = 0$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property: template $b^m d^p b^m d^p$ hence every pair of non-adjacent segments have equal length.

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$ $b^m d^p b^m d^p$ where $m > \pi$, $p > \pi$

w_0 belongs to L because it is obtained from the template

w_0 must pump because $|w_0| = 2m + 2p > 4\pi > \pi$

In any "pumping" decomposition of w_0, the pumping window satisfies the following property: located within a single segment or a pair of adjacent segments

because its length is $\leq \pi$, so it is too long to extend through 3 segments.

By pumping 1 times, we obtain a string w_1

which does not belong to L because in w_1 pumping has increased the length of two adjacent segment, so at least one pair of non-adjacent segments have different length.

Since L violates the Pumping Lemma, it is not context free.

LAST NAME:

FIRST NAME:

Problem 6 [10 points]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$c^\ell a^p b^\ell a^p$

$$c^i b^j a^k b^\ell c^m d^n a^p d^q$$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = \ell$, $k = p$, $j = 0$, $q = 0$, $m = 0$, $n = 0$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property: template $c^\ell a^p b^\ell a^p$ so # symbols in non-adjacent segments is equal.

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$ $c^\ell a^p b^\ell a^p$ where $\ell > \pi$, $p > \pi$

w_0 belongs to L because it is obtained from the template

w_0 must pump because $|w| = 2\ell + 2p > 4\pi > \pi$

In any "pumping" decomposition of w_0, the pumping window satisfies the following property: located within either a single segment or two adjacent segments

because its length is $\leq \pi$ while each segment is longer than π, so it cannot extend to 3 segments!

By pumping 1 times, we obtain a string w_1

which does not belong to L because in w_1 in some non-adjacent segment pair pumping has introduced symbols into one segment only.

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 6 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$c^i b^j c^k b^\ell a^m d^n a^p d^q$$

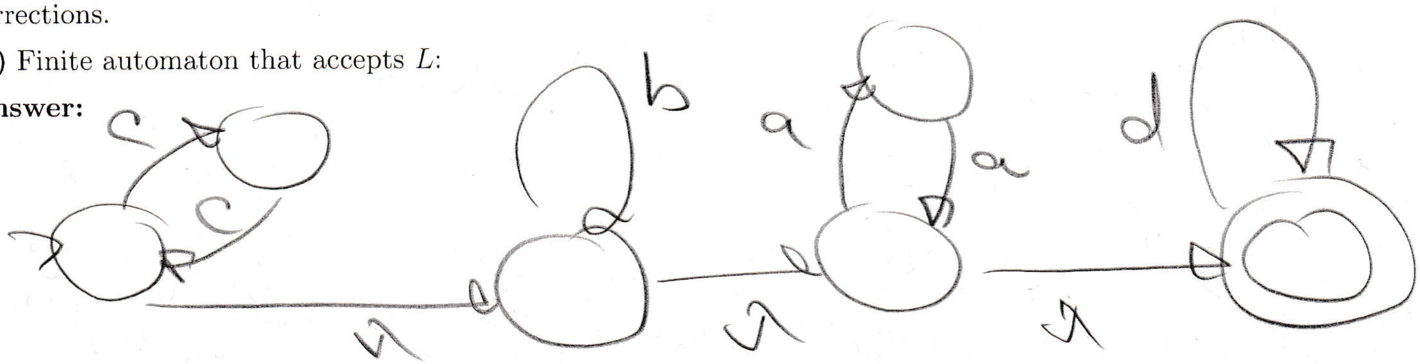
$$c^k c^k b^\ell a^p a^p d^q = c^{2k} b^\ell a^{2p} d^q$$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = k,\ j = 0,\ m = p,\ n = 0$.

If L is regular, then use part (a) of the answer space below to draw a state transition graph of a finite automaton that accepts L, and do not write anything in part (b). If L is not regular, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not regular. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Finite automaton that accepts L:

Answer:



(b) Proof that L is not regular:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is regular. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string which does not belong to L because

Since L violates the Pumping Lemma, it is not regular.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 4 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$c^i b^j a^k b^\ell c^m d^n a^p d^q$$

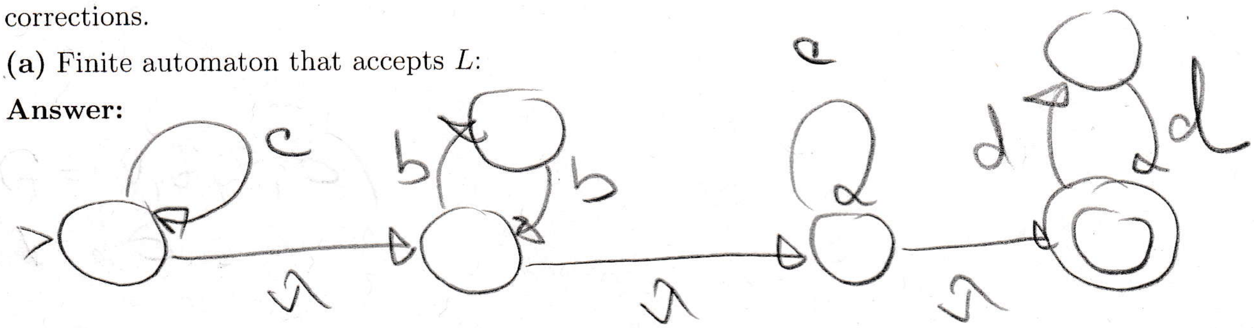
$$c^i b^l b^l c^m d^q d^q = c^i b^{2l} c^m d^{2q}$$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $q = n,\ k = 0,\ j = \ell,\ p = 0$.

If L is regular, then use part (a) of the answer space below to draw a state transition graph of a finite automaton that accepts L, and do not write anything in part (b). If L is not regular, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not regular. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Finite automaton that accepts L:

Answer:



(b) Proof that L is not regular:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is regular. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string

which does not belong to L because

Since L violates the Pumping Lemma, it is not regular.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 5 [10 points]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$a^i b^j c^k a^\ell d^m b^n c^p d^q$$

$a^m b^k c^k d^m b^p c^p$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = m,\ j = k,\ \ell = 0,\ n = p,\ q = 0$.

If L is regular, then use part (a) of the answer space below to draw a state transition graph of a finite automaton that accepts L, and do not write anything in part (b). If L is not regular, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not regular. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Finite automaton that accepts L:

Answer:

(b) Proof that L is not regular:

Observe that all words of L satisfy the following characteristic property: template:

$a^m b^k c^k d^m b^p c^p$, so #a's = #d's

Assume the opposite, that L is regular. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$ $a^n d^n$ where $n > \pi$

w_0 belongs to L because obtained from template with $k = 0,\ p = 0$

w_0 must pump because $|w_0| = 2n > \pi$

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

is equal to a^j for some $j > 0$

because it is located within first π symbols, $\pi < n$

By pumping 1 times, we obtain a string $w_1 = a^{n+j} d^n$

which does not belong to L because

#a's $= n + j \neq n =$ #d's, since $j > 0$

Since L violates the Pumping Lemma, it is not regular.

LAST NAME:

FIRST NAME:

Problem 5 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$a^i c^j b^k a^\ell d^m c^n b^p d^q$$

$\underbrace{c^k b^k}\, a^q \underbrace{d^n c^n}\, d^q$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $m = n,\ j = k,\ i = 0,\ \ell = q,\ p = 0$.

If L is regular, then use part (a) of the answer space below to draw a state transition graph of a finite automaton that accepts L, and do not write anything in part (b). If L is not regular, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not regular. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Finite automaton that accepts L:

Answer:

(b) Proof that L is not regular:

Observe that all words of L satisfy the following characteristic property: template

$c^k b^k a^q d^n c^n d^q$, #b's = #c's to the left of b's

Assume the opposite, that L is regular. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$ $c^k b^k$ where $k > \pi$

w_0 belongs to L because it is obtained from the template with $q = 0, n = 0$.

w_0 must pump because $|w_0| = 2k > 2\pi > \pi$

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

is equal to c^j for some $j > 0$

because it is located within first π symbols, and $\pi < k$.

By pumping 1 times, we obtain a string $w_1 = c^{k+j} b^k$

which does not belong to L because

#c's = $k+j \neq k$ = #b's

Since L violates the Pumping Lemma, it is not regular.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 1 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$b^m d^k c^k a^\ell c^m b^n a^p d^p$ $b^i d^j c^k a^\ell c^m b^n a^p d^q$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = m$, $j = k$, $p = q$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

$G = (V, \Sigma, P, S)$

$\Sigma = \{a, b, c, d\}$

$V = \{S, A, B, D, E, F\}$

P:
$S \to ABD$
$A \to bAc \mid EF$
$E \to dEc \mid \lambda$
$F \to aF \mid \lambda$
$B \to bB \mid \lambda$
$D \to aDd \mid \lambda$

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string which does not belong to L because

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 2 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$d^q c^k b^k c^p a^n b^n d^p a^q$

$$d^i c^j b^k c^\ell a^m b^n d^p a^q$$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = q,\ j = k,\ \ell = p,\ m = n$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

$G = (V, \Sigma, P, S)$

$\Sigma = \{a, b, c, d\}$

$V = \{S, A, B, D\}$

P: $S \to dSa \mid AB$

$A \to cAb \mid \lambda$

$B \to cBd \mid D$

$D \to aDb \mid \lambda$

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string

which does not belong to L because

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 3 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$m+q = q+m$

$$b^i a^j d^k c^\ell d^m b^n c^p a^q$$

$b^m a^j d^q d^m b^p c^p a^q = b^m a^j d^m d^q b^p c^p a^q$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = m$, $k = q$, $\ell = 0$, $p = n$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

$G = (V, \Sigma, P, S)$

$\Sigma = \{a, b, c, d\}$

$V = \{S, A, B, D, E\}$

P: $S \to AB$

$A \to bAd \mid D$

$D \to aD \mid \lambda$

$B \to dBa \mid E$

$E \to bEc \mid \lambda$

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string which does not belong to L because

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 2 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$a^{q} c^{m} b^{\ell} c^{\ell} d^{m} b^{p} a^{p} d^{q}$ $\quad a^i c^j b^k c^\ell d^m b^n a^p d^q$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = q,\ j = m,\ k = \ell,\ n = p$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

$G = (V, \Sigma, P, S)$

$V = \{S, A, B, D\}$

$\Sigma = \{a, b, c, d\}$

$P:$
$S \to aSd \mid AB$
$A \to cAd \mid D$
$D \to bDc \mid \lambda$
$B \to bBa \mid \lambda$

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string which does not belong to L because

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: __________

FIRST NAME: __________

Problem 3 [**10 points**]

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$d^i b^j c^k a^\ell c^m d^n a^p b^q$$

$\underbrace{d^i b^j}\ c^k\ a^\ell\ c^m\ \underbrace{d^p a^p}\ b^\ell$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = j,\ n = p,\ \ell = q$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

$G = (V, \Sigma, P, S)$

$V = \{S, A, B, D, E\}$

$\Sigma = \{a, b, c, d\}$

P: $S \to ABD$

$A \to dAb \mid \Lambda$

$B \to cB \mid \Lambda$

$D \to aDb \mid BE$

$E \to dEa \mid \Lambda$

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string

which does not belong to L because

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 4 [**10 points**]

$\boxed{m+q = q+m}$

Let L be the language over the alphabet $\Sigma = \{a, b, c, d\}$ that contains exactly those strings whose form is:

$$c^i b^j a^k d^\ell a^m c^n d^p b^q,$$

$$c^m b^j a^q a^m c^p d^p b^q = c^m b^j a^m a^q c^p d^p b^q$$

where $i, j, k, \ell, m, n, p, q \geq 0$ are natural numbers such that: $i = m$, $k = q$, $\ell = 0$, $p = n$.

If L is context free, then use part (a) of the answer space below to write a complete formal definition of a context free grammar that generates L, and do not write anything in part (b). If L is not context free, then do not write anything in part (a) of the answer space, but complete the missing parts of the text given in part (b) so as to obtain a proof that L is not context free. If the text given in part (b) requires corrections (in addition to completion), then make these corrections.

(a) Grammar that generates L:

Answer:

$G = (V, \Sigma, P, S)$

$V = \{S, A, B, D, E\}$

$\Sigma = \{a, b, c, d\}$

P: $S \to AB$

$A \to cAa \mid D$

$D \to bD \mid \lambda$

$B \to aBb \mid E$

$E \to cEd \mid \lambda$

(b) Proof that L is not context free:

Observe that all words of L satisfy the following characteristic property:

Assume the opposite, that L is context free. Let π be the constant as in the Pumping Lemma for L. Let $w_0 \in L$ be a string defined as follows:

$w_0 =$

w_0 belongs to L because

w_0 must pump because

In any "pumping" decomposition of w_0, the pumping window satisfies the following property:

because

By pumping times, we obtain a string

which does not belong to L because

Since L violates the Pumping Lemma, it is not context free.

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 7 [**20 points**]

Let L be the language accepted by the pushdown automaton $M = (Q, \Sigma, \Gamma, \delta, q, F)$ where: $Q = \{q, p, s, t\}$, $\Sigma = \{a, b, c, d\}, \Gamma = \{A, B, E, F, R\}$, $F = \{t\}$ and the transition function δ is defined as follows:

$[q, \lambda, \lambda, p, FREE]$	$[s, c, \lambda, s, A]$	$[t, a, E, t, \lambda]$
$[p, d, \lambda, p, B]$	$[s, \lambda, \lambda, t, \lambda]$	$[t, b, F, t, \lambda]$
$[p, \lambda, \lambda, s, \lambda]$	$[t, a, A, t, \lambda]$	$[t, c, R, t, \lambda]$
	$[t, b, B, t, \lambda]$	

(Recall that M is defined so as to accept by final state and empty stack. Furthermore, if an arbitrary stack string, say $X_1 \ldots X_n \in \Gamma^*$ where $n \geq 2$, is pushed on the stack by an individual transition, then the leftmost symbol X_1 is pushed first, while the rightmost symbol X_n is pushed last.)

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following two symbols:

1 if the string next to the rectangle belongs to L;
0 if the string next to the rectangle belongs to $\overline{L}$;

s	$s \in L$
λ	0
$aacb$	1
$bcaa$	0
$caaacb$	1
$ccaaaacb$	1
$dbaacb$	1
$dbac$	0
$dbacaacb$	0
$dcab$	0
$dcabaacb$	1

(b) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

Advice: template

$d^n c^k a^k b^n aacb$

(c) Is the language $L \cap (\boldsymbol{ab} \cup \boldsymbol{cbc})^*$ context free? Explain your answer.

Answer: yes. Intersection of a context-free language with regular is context-free.

(d) Is L decidable? Explain your answer.

Answer: Yes. All context free languages are decidable.

(e) State the cardinality of L. If L is finite, state the exact number; if L is infinite, specify whether it is countable or not countable.

Answer: infinite and countable.

Answer:

$G = (V, \Sigma, P, S)$

$V = \{S, A, B\}$

$\Sigma = \{a, b, c, d\}$

$P: S \to Aaacb$

$A \to dAb \mid B$

$B \to cBa \mid \lambda$

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 7 [**20 points**]
Let L be the language accepted by the pushdown automaton $M = (Q, \Sigma, \Gamma, \delta, q, F)$ where: $Q = \{q, p, s, t\}$, $\Sigma = \{a, b, c, d\}, \Gamma = \{A, B, D, P, S\}$, $F = \{t\}$ and the transition function δ is defined as follows:

$[q, \lambda, \lambda, p, PASS]$	$[s, c, \lambda, s, D]$	$[t, a, A, t, \lambda]$
$[p, a, \lambda, p, B]$	$[s, \lambda, \lambda, t, \lambda]$	$[t, b, P, t, \lambda]$
$[p, \lambda, \lambda, s, \lambda]$	$[t, d, D, t, \lambda]$	$[t, c, S, t, \lambda]$
	$[t, b, B, t, \lambda]$	

(Recall that M is defined so as to accept by final state and empty stack. Furthermore, if an arbitrary stack string, say $X_1 \ldots X_n \in \Gamma^*$ where $n \geq 2$, is pushed on the stack by an individual transition, then the leftmost symbol X_1 is pushed first, while the rightmost symbol X_n is pushed last.)

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following two symbols:

1 if the string next to the rectangle belongs to L;
0 if the string next to the rectangle belongs to $\overline{L}$;

s	$s \in L$
λ	0
$abccab$	1
$abcd$	0
$abcdccab$	0
$acdb$	0
$acdbccab$	1
$bacc$	0
$ccab$	1
$ccddccab$	1
$cdccab$	1

(b) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:
Advice: template:
$a^n c^k d^k b^n ccab$

Answer:
$G = (V, \Sigma, P, S)$
$V = \{S, A, B\}$
$\Sigma = \{a, b, c, d\}$

P: $S \to Accab$
$A \to aAb \mid B$
$B \to cBd \mid \lambda$

(c) Is the language $L \cap (\mathbf{ab} \cup \mathbf{cbc})^*$ context free? Explain your answer.

Answer: Yes. Intersection of context-free language with regular language is context-free.

(d) Is L decidable? Explain your answer.

Answer: Yes Every context free language is decidable.

(e) State the cardinality of L. If L is finite, state the exact number; if L is infinite, specify whether it is countable or not countable.

Answer:
infinite and countable.

LAST NAME: ____________________

FIRST NAME: ____________________

Problem 8 [20 points]
Let L be the language accepted by the pushdown automaton $M = (Q, \Sigma, \Gamma, \delta, q, F)$ where: $Q = \{q, p, s, t, v\}$, $\Sigma = \{a, b, c, d\}, \Gamma = \{B, D, E, F, R\}$, $F = \{v\}$ and the transition function δ is defined as follows:

$$\begin{array}{lll} [q, \lambda, \lambda, p, FREE] & [s, b, B, s, \lambda] & [v, a, E, v, \lambda] \\ [p, a, \lambda, p, B] & [s, \lambda, E, t, E] & [v, b, F, v, \lambda] \\ [p, \lambda, \lambda, s, \lambda] & [t, d, \lambda, t, D] & [v, c, D, v, \lambda] \\ & [t, \lambda, \lambda, v, \lambda] & [v, d, R, v, \lambda] \end{array}$$

(Recall that M is defined so as to accept by final state and empty stack. Furthermore, if an arbitrary stack string, say $X_1 \ldots X_n \in \Gamma^*$ where $n \geq 2$, is pushed on the stack by an individual transition, then the leftmost symbol X_1 is pushed first, while the rightmost symbol X_n is pushed last.)

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following two symbols:

1 if the string next to the rectangle belongs to L;
0 if the string next to the rectangle belongs to $\overline{L}$;

s	$s \in L$
λ	0
$aabbaadb$	1
$aadb$	1
$abaadb$	1
$abab$	0
$abdcaadb$	1
$acbdaadb$	0
$bdaa$	0
$dbaa$	0
$dcaadb$	1

(b) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

Advice: template
$a^n b^n d^k c^k aadb$

Answer:
$G = (V, \Sigma, P, S)$
$V = \{S, A, B\}$
$\Sigma = \{a, b, c, d\}$

P:
$S \to ABaadb$
$A \to aAb \mid \lambda$
$B \to dBc \mid \lambda$

(c) Is L recursively enumerable? Explain your answer.

Answer: Yes. Every context-free language is rec. en.

(d) For a recursively enumerable language G, let the property $P_1(G)$ be defined as follows:

$$P_1(G) \iff G \subseteq L$$

Is P_1 a non-trivial property of recursively enumerable languages? Explain your answer.

Answer: Yes, by definition, since it is true for L and false for Σ^*

(e) State the value of $P_1(\emptyset)$.

Answer:

1

(f) State the value of $P_1(\Sigma^*)$.

Answer:

0

(g) Explain how to construct an algorithm that solves the following problem:

INPUT: String w over Σ.

OUTPUT: **yes** if w is a string which belongs to the set of exactly those strings on which the ~~Turing Machine~~ pda M (defined at the beginning of this problem) ~~diverges~~ accepts;
no otherwise.

If this algorithm does not exist, prove it.

Answer:

Simulate M (or any other automaton that decides L) and do as it does.

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 8 [**20 points**]

Let L be the language accepted by the pushdown automaton $M = (Q, \Sigma, \Gamma, \delta, q, F)$ where: $Q = \{q, p, s, t, v\}$, $\Sigma = \{a, b, c, d\}, \Gamma = \{A, B, D, P, S\}$, $F = \{v\}$ and the transition function δ is defined as follows:

$$\begin{array}{lll}
[q, \lambda, \lambda, p, PASS] & [s, b, B, s, \lambda] & [v, a, P, v, \lambda] \\
[p, a, \lambda, p, B] & [s, \lambda, S, t, S] & [v, b, A, v, \lambda] \\
[p, \lambda, \lambda, s, \lambda] & [t, c, \lambda, t, D] & [v, c, S, v, \lambda] \\
 & [t, \lambda, \lambda, v, \lambda] & [v, d, D, v, \lambda]
\end{array}$$

(Recall that M is defined so as to accept by final state and empty stack. Furthermore, if an arbitrary stack string, say $X_1 \ldots X_n \in \Gamma^*$ where $n \geq 2$, is pushed on the stack by an individual transition, then the leftmost symbol X_1 is pushed first, while the rightmost symbol X_n is pushed last.)

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following two symbols:

1 if the string next to the rectangle belongs to L;
0 if the string next to the rectangle belongs to $\overline{L}$;

s	$s \in L$
λ	
$aabbccba$	
$abab$	
$abccba$	
$abcdccba$	
$acbcdcba$	
$bdaa$	
$ccba$	
$cdccba$	
$dbaa$	

(b) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

Advice: template:

$a^n b^n c^k d^k ccba$

Answer: $G = (V, \Sigma, P, S)$

$V = \{S, A, B\}$

$\Sigma = \{a, b, c, d\}$

P: $S \to ABccba$

$A \to aAb \mid \lambda$

$B \to cBd \mid \lambda$

(c) Is L recursively enumerable? Explain your answer.

Answer: Yes. All context-free languages are r.e.

(d) For a recursively enumerable language G, let the property $P_1(G)$ be defined as follows:

$$P_1(G) \iff L \subseteq G$$

Is P_1 a non-trivial property of recursively enumerable languages? Explain your answer.

Answer: Yes, as it is true for L and false for $\emptyset$

(e) State the value of $P_1(\emptyset)$.

Answer:

0

(f) State the value of $P_1(\Sigma^*)$.

Answer:

1

(g) Explain how to construct an algorithm that solves the following problem:

INPUT: String w over Σ.

OUTPUT: **yes** if w is a string which belongs to the set of exactly those strings on which the pda ~~Turing Machine~~ M (defined at the beginning of this problem) ~~diverges;~~ accepts

no otherwise.

If this algorithm does not exist, prove it.

Answer:

Simulate M (or whichever machine decides L) and do as it does

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 9 [20 points]
Consider the Turing machine $M = (Q, \Sigma, \Gamma, \delta, q, F)$ such that: $Q = \{q, p, s, e\}$; $\Sigma = \{a, b, c\}$; $\Gamma = \{B, a, b, c\}$; $F = \{s\}$; and δ is defined by the following transition set:

d

$$
\begin{array}{lll}
[q,a,p,a,R] & [p,a,s,a,R] & [s,a,q,a,R] \\
[q,b,p,b,R] & [p,b,s,b,R] & [s,b,q,b,R] \\
[q,c,p,c,R] & [p,c,s,c,R] & [s,c,q,c,R] \\
[q,B,e,B,R] & [e,B,e,B,R] & \\
[e,a,e,a,R] & [e,b,e,b,R] & [e,c,e,c,R]
\end{array}
$$

(M has an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

Let L_A be the set of string which M *accepts.*
Let L_R be the set of string which M *rejects.*
Let L_D be the set of string on which M *diverges.*

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following ~~two~~ symbols:

A if the string next to the rectangle belongs to L_A;
R if the string next to the rectangle belongs to L_R;
D if the string next to the rectangle belongs to L_D;

s	
λ	D
a	R
aa	A
aab	D
$aadb$	A
abd	A
bc	A
$cbab$	D
cc	A
$cdbd$	R

(b) Write a regular expression that defines L_A. If such a regular expression does not exist, prove it.

Answer: $(\sigma\sigma\sigma)^*\sigma\sigma(\lambda \cup d(d \cup \sigma)^*)$
where $\sigma = (a \cup b \cup c)$

(c) Write a regular expression that defines L_R. If such a regular expression does not exist, prove it.

Answer: $(\sigma\sigma\sigma)^* d(d \cup \sigma)^*$
$\cup$
$(\sigma\sigma\sigma)^*\sigma(\lambda \cup d(d \cup \sigma)^*)$
where $\sigma = (a \cup b \cup c)$

(d) Write a regular expression that defines L_D. If such a regular expression does not exist, prove it.

Answer: $(\sigma\sigma\sigma)^*$
where $\sigma = (a \cup b \cup c)$

(e) Which language (if any) is decided by M? Explain your answer.

Answer: None. M cannot decide because on some inputs M does not halt.

(f) Is L_A recursively enumerable? Explain your answer.

Answer: Yes. M accepts it.

(g) Is L_D decidable? Explain your answer.

Answer: Yes, $L_D = \overline{L_A \cup L_R}$ hence regular (since L_A, L_R are regular by (b), (c)) and decidable.

LAST NAME:

FIRST NAME:

Problem 9 [**20 points**]
Consider the Turing machine $M = (Q, \Sigma, \Gamma, \delta, q, F)$ such that: $Q = \{q, p, s, e\}$; $\Sigma = \{a, b, c\}$; $\Gamma = \{B, a, b, c\}$; $F = \{q\}$; and δ is defined by the following transition set:

$[q, a, p, a, R]$	$[p, a, s, a, R]$	$[s, a, q, a, R]$
$[q, b, p, b, R]$	$[p, b, s, b, R]$	$[s, b, q, b, R]$
$[q, c, p, c, R]$	$[p, c, s, c, R]$	$[s, c, q, c, R]$
$[e, B, e, B, R]$	$[p, B, e, B, R]$	
$[e, a, e, a, R]$	$[e, b, e, b, R]$	$[e, c, e, c, R]$

(M has an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

Let L_A be the set of string which M *accepts*.
Let L_R be the set of string which M *rejects*.
Let L_D be the set of string on which M *diverges*.

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following ~~two~~ symbols:

A if the string next to the rectangle belongs to L_A;
R if the string next to the rectangle belongs to L_R;
D if the string next to the rectangle belongs to L_D;

s	
λ	A
aa	R
aab	A
$abcb$	D
$acad$	A
b	D
bad	R
cc	R
db	A
$ddca$	A

(b) Write a regular expression that defines L_A. If such a regular expression does not exist, prove it.

Answer: $(\sigma\sigma\sigma)^*(\lambda \cup d(d\cup\sigma)^*)$

$\cup$

$d(d\cup\sigma)^*$

where $\sigma = (a\cup b\cup c)$

(c) Write a regular expression that defines L_R. If such a regular expression does not exist, prove it.

Answer: $(\sigma\sigma\sigma)^*\sigma\sigma(\lambda\cup(d\cup\sigma)^*)$

$\cup$

$(\sigma\sigma\sigma)^*\sigma(\lambda\cup(d\cup\sigma)^*)$

where $\sigma = (a\cup b\cup c)$

(d) Write a regular expression that defines L_D. If such a regular expression does not exist, prove it.

Answer:

$(\sigma\sigma\sigma)^*\sigma$

where $\sigma = (a\cup b\cup c)$

(e) Which language (if any) is decided by M? Explain your answer.

Answer: None. M cannot decide because it does not halt on some inputs

(f) Is L_A recursively enumerable? Explain your answer.

Answer: Yes. M accepts it.

(g) Is L_D decidable? Explain your answer.

Answer: Yes. It is the complement of regular language $L_A \cup L_R$ (parts b, c) so regular & decidable

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 10 [**20 points**]

Consider the Turing machine

$M = (Q, \Sigma, \Gamma, \delta, q, F)$ such that: $Q = \{q, p, v, x, y, z\}$ e

$\Sigma = \{a, b, c, d\}$; $\Gamma = \{B, a, b, c, d, N\}$; $F = \{z\}$; and δ is defined by the following transition set:

$[q, a, p, N, R]$	$[p, a, p, a, R]$	$[v, a, x, a, L]$
$[q, b, p, N, R]$	$[p, b, p, b, R]$	$[v, b, x, b, L]$
$[q, d, p, N, R]$	$[p, c, v, c, L]$	$[v, d, x, d, L]$
$[q, B, e, B, R]$	$[p, d, p, d, L]$	
		$[x, d, y, b, L]$
$[e, a, e, a, R]$		$[y, b, z, a, L]$
$[e, b, e, b, R]$		
$[e, c, e, c, R]$		
$[e, B, e, B, R]$		

(M has an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

Let L_A be the set of string which M *accepts*.
Let L_R be the set of string which M *rejects*.
Let L_D be the set of string on which M *diverges*.

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following two symbols:

A if the string next to the rectangle belongs to L_A;
R if the string next to the rectangle belongs to L_R;
D if the string next to the rectangle belongs to L_D;

s	
λ	D
$abcc$	R
ac	R
ad	R
$badbadac$	D
$bbaaddcc$	D
bdc	D
c	R
aa	R
$ccadc$	D

(b) Write a regular expression that defines L_A. If such a regular expression does not exist, prove it.

Answer: $\emptyset$

Advice: It looks for $bd(a \cup b \cup d)c$ but diverges (in place) at bd!

(c) Write a regular expression that defines L_D. If such a regular expression does not exist, prove it.

Answer:

$\lambda \cup (a \cup b \cup c)(a \cup b)^*(a \cup b)d(a \cup b \cup c \cup d)$

(d) Explain how to construct an algorithm that solves the following problem:

INPUT: String w over Σ.

OUTPUT: **yes** if w represents a Turing Machine that accepts exactly those strings which the Turing Machine M (defined at the beginning of this problem) accepts;
no otherwise.

If this algorithm does not exist, prove it.

Answer: Impossible by Rice's Th. The algorithm would decide if $L(w)$ has the nontrivial property "$= L(M)$"

(e) Explain how to construct a machine that operates as follows:

INPUT: String w over Σ.

OUTPUT: **halt** if w is a string which belongs to the set of exactly those strings on which the Turing Machine M (defined at the beginning of this problem) halts;
diverge otherwise.

If this machine does not exist, prove it.

Answer:

The required machine is the machine M defined at the start of this problem.

LAST NAME: ______________________

FIRST NAME: ______________________

Problem 10 [20 points]
Consider the Turing machine $M = (Q, \Sigma, \Gamma, \delta, q, F)$ such that: $Q = \{q, p, v, x, y, z, e\}$; $\Sigma = \{a, b, c, d\}$; $\Gamma = \{B, a, b, c, d, N\}$; $F = \{z\}$; and δ is defined by the following transition set:

$[q, a, p, N, R]$	$[p, a, p, a, R]$	$[v, a, x, a, L]$
$[q, b, p, N, R]$	$[p, b, p, b, R]$	$[v, b, x, b, L]$
$[q, c, p, N, R]$	$[p, c, p, c, L]$	$[v, c, x, c, L]$
$[q, B, e, B, R]$	$[p, d, v, d, L]$	
		$[x, b, y, b, L]$
$[e, a, e, a, R]$		$[y, a, z, a, L]$
$[e, b, e, b, R]$		
$[e, c, e, c, R]$		
$[e, B, e, B, R]$		

(M has an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

Let L_A be the set of string which M *accepts*.
Let L_R be the set of string which M *rejects*.
Let L_D be the set of string on which M *diverges*.

(a) In the table below, fill the ten empty rectangles by writing into each rectangle one of the following ~~two~~ symbols:

A if the string next to the rectangle belongs to L_A;
R if the string next to the rectangle belongs to L_R;
D if the string next to the rectangle belongs to L_D;

s	
λ	D
abcc	D
abd	R
ad	R
babadbaba	A
c	R
cabd	R
ccaabbdd	R
cd	R
dd	R

(b) Write a regular expression that defines L_A. If such a regular expression does not exist, prove it.

Answer: $(a \cup b \cup c)(a \cup b)^* ab(a \cup b)d(a \cup b \cup c \cup d)^*$

(c) Write a regular expression that defines L_D. If such a regular expression does not exist, prove it.

Answer: $\lambda \cup (a \cup b \cup c)(a \cup b)^*(a \cup b)c(a \cup b \cup c \cup d)^*$

(d) Explain how to construct a machine that operates as follows:

INPUT: String w over Σ.

OUTPUT: **halt** if w is a string which belongs to the set of exactly those strings on which the Turing Machine M (defined at the beginning of this problem) halts;
diverge otherwise.

If this machine does not exist, prove it.

Answer: Machine M (defined at the start of this problem) is the machine.

(e) Explain how to construct an algorithm that solves the following problem:

INPUT: String w over Σ.

OUTPUT: **yes** if w represents a Turing Machine that accepts exactly those strings which the Turing Machine M (defined at the beginning of this problem) accepts;
no otherwise.

If this algorithm does not exist, prove it.

Answer: Impossible by Rice's Th. The algorithm would decide whether L(w) has a nontrivial property "= L(M)"